

Head Motion Stabilization During Quadruped Robot Locomotion: Combining Dynamical Systems and a Genetic Algorithm

Cristina P Santos, Miguel Oliveira, Ana Maria A. C. Rocha and Lino Costa

Abstract—The head shaking that results from robot locomotion is important because it difficult stable image acquisition and the possibility to rely on that information to act accordingly, for instance, to achieve visually-guided locomotion.

In this article, we focus on the development of a head controller able to minimize the head motion induced by locomotion itself. Specifically, we propose a combined approach to generate head movement stabilization on a quadruped robot, using Central Pattern Generators (CPGs) and a genetic algorithm.

Head movement is generated by CPGs which are modelled as autonomous differential equations. This approach allows to explicitly specify parameters such as amplitude, offset and frequency of movement and to smoothly modulate the generated trajectories according to changes in these parameters. It is therefore easy to combine the CPG with an optimization method. A genetic algorithm determines the best set of parameters that generates the head movement that reduces the head shaking caused by locomotion.

Experimental results on a simulated AIBO robot demonstrate that the proposed approach generates head movement that does not eliminate but reduces the one induced by locomotion.

I. INTRODUCTION

Robot locomotion is a challenging task that involves several relevant subtasks, not yet completely solved. The motion of quadruped, biped and snake-like robots, for instance, with cameras mounted in their heads, causes head shaking. This kind of disturbances, generated by locomotion itself, makes it difficult to keep the visual frame stable and, therefore, to act according to the visual information. On the contrary, their counter part biological mechanisms are able to stabilize the head movement, even when they change the type of gait or adapt to the terrain.

As a basic research to realize visually-guided quadruped locomotion, we aim in this article at head stabilization of a walking quadruped robot. In our research, we propose a motion stabilization system of an ers-7 AIBO quadruped robot, which performs its own head motion according to a feedforward controller. Several similar works have been proposed in literature [5], [8], [7], [6]. But these methods consider that the robot moves according to a scheduled robot motion plan, which implies that space and time constraints on robot motion must be known before hand as well as robot and environment models. As such, control is based on this scheduled plan. Other works have successfully achieved gaze

stabilization [6], that consists on image stabilization during head movements in space. However, binocular vision systems were considered, requiring additional hardware in the robot.

In this work, we propose a combined approach to generate head movement stabilization on a quadruped robot, using Central Pattern Generators (CPGs) and a genetic algorithm.

We propose a head controller, based on CPGs, that generates trajectories for tilt, pan and nod head joints. CPGs are neural networks located in the spine of vertebrates, able to generate coordinated rhythmic movements, namely locomotion [12]. These CPGs are modelled as coupled oscillators and solved using numeric integration. They have been applied in drumming [1] and postural control [4].

This dynamical systems approach model for CPGs presents multiple interesting properties comparatively to other methods [13]. These include: low computation cost which is well-suited for real time; the stability properties of the limit cycle behavior (i.e. perturbations are quickly forgotten); intrinsic robustness against small perturbations; the smooth online modulation of trajectories through changes in the dynamical systems parameters and phase-locking between the different oscillators for different DOFs.

The proposed CPG, based on Hopf oscillators, allows to explicitly specify parameters such as amplitude, offset and frequency of movement and to smoothly modulate the generated trajectories according to changes in these parameters. In order to achieve the desired head movement, opposed to the one induced by locomotion, it is necessary to appropriately tune the CPG parameters. This is achieved by optimizing the CPG parameters using an optimization method. Optimization is done off-line according to the head movement induced by the locomotion when no stabilization procedure was performed.

This optimization is a non-linear problem where continuity and convexity conditions are not guarantee. Thus, searching for a global optimum is a difficult task that requires approaches based on stochastic algorithms like evolutionary algorithms, in particular, genetic algorithms. These are search algorithms that mimic the process of natural selection [2]. Thus, unlike conventional algorithms, in general, only the information regarding the objective to optimize is required. Moreover, they are based on a population that evolves over time, possibly in the direction of the optimum.

This article is structured as follows: In Section II, we describe the system architecture and we explain how we generate locomotion and head movement. Section III presents the optimization process and some results are discussed. Simulated results are described in Section IV. The paper ends

Cristina Santos and Miguel Oliveira are with Industrial Electronics Department, School of Engineering, University of Minho, 4800-058 Guimarães, Portugal cristina@dei.uminho.pt, mcampos@dei.uminho.pt

Ana Rocha and Lino Costa are with Production Systems Department, School of Engineering, University of Minho, 4800-058 Guimarães, Portugal arocha@dps.uminho.pt, lac@dps.uminho.pt

with discussion and conclusions in Section V.

II. SYSTEM ARCHITECTURE

Our aim is to propose a control architecture that is able to generate locomotion for a quadruped robot and to generate head motion such as to minimize the head movement induced by the locomotion itself.

The overall system architecture is depicted in fig. 1.

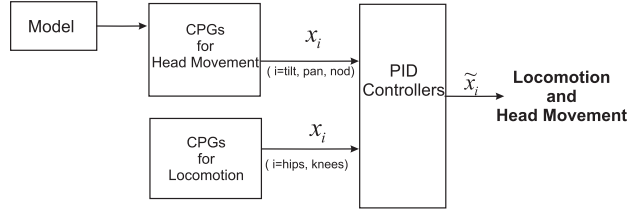


Fig. 1. Overall system architecture

Movement controllers are biologically inspired in the concept of CPGs. A locomotion controller generates hip and knee trajectories. A head controller specifies the planned neck tilt, pan and nod joint values. These trajectories are used as input for the PID controllers of these joints.

The head controller parameters have to be tuned such that the resultant movement is as desired. Using our CPG approach allows us to assign explicit parameters for each of the nonlinear oscillators, independently controlling the amplitude, offset and frequency of the movement. We apply a stochastic optimization method, a Genetic Algorithm (GA), in order to determine the best set of parameters that results in, or close to the desired movement. This set of parameters constitute the Model module in Fig. 1.

A. Locomotion Generation

In this section we present the network of CPGs used to generate locomotion. A CPG for a given degree-of-freedom (DOF) is modelled as coupled Hopf oscillators, that generate a rhythmic movement.

1) *Rhythmic Movement Generation*: Rhythmic movements are generated by the following Hopf oscillator

$$\dot{x}_i = \beta (\mu_i - r_i^2) (x_i - O_i) - \omega z_i, \quad (1)$$

$$\dot{z}_i = \beta (\mu_i - r_i^2) z_i + \omega (x_i - O_i), \quad (2)$$

where $r_i = \sqrt{(x_i - O_i)^2 + z_i^2}$, amplitude of the oscillations are given by $R = \sqrt{\mu_i}$, ω specifies the oscillations frequency (in rad s⁻¹) and relaxation to the limit cycle is given by $\frac{1}{2\beta\mu_i}$.

This Hopf oscillator contains a bifurcation from a stable fixed point at $x_i = O_i$ (when $\mu_i < 0$) to a structurally stable, harmonic limit cycle, for $\mu_i > 0$. The fixed point x_i has an offset given by O_i .

We motivate the choice of an Hopf oscillator because it can be completely analytically solved, which facilitates the specification of parameters such as frequency, amplitude or offset. Further, it is possible to achieve smooth modulation

of the generated trajectories according to changes in these parameters. This is interesting for trajectory generation in a robot. In [1] it was shown how the generated trajectories can easily and smoothly be modulated by modifying on the fly the offset values (O becomes the state variable of another dynamical system).

This Hopf oscillator describes a rhythmic motion which amplitude of movement is specified by μ_i , offset by O_i and its frequency by ω .

2) *Locomotion Controller Architecture*: We have to couple the oscillators in order to ensure phase-locked synchronization between the hip and knee DOFs of the robot, and generate locomotion with the desired gait.

Fig. 2 depicts the network structure used to generate the walking gait. Hopf oscillators of the hips are bilaterally coupled, these couplings being illustrated by right-left arrows, and hip Hopf oscillators are unilaterally coupled to the corresponding knee Hopf oscillators. For the hip joints, this is achieved by modifying (1) and (2) as follows:

$$\begin{aligned} \begin{bmatrix} \dot{x}_{i[1]} \\ \dot{z}_{i[1]} \end{bmatrix} &= \begin{bmatrix} \beta\mu_i & \omega \\ -\omega & \beta\mu_i \end{bmatrix} \begin{bmatrix} x_{i[1]} - O_{i[1]} \\ z_{i[1]} \end{bmatrix} - \beta r_{i[1]}^2 \begin{bmatrix} x_{i[1]} - O_{i[1]} \\ z_{i[1]} \end{bmatrix} \\ &+ \sum_{j \neq i} \mathbf{R}(\theta_{i[1]}^{j[1]}) \begin{bmatrix} x_{j[1]} - O_{j[1]} \\ z_{j[1]} \end{bmatrix} \end{aligned}$$

For the knee joints, we modify (1) and (2) as follows:

$$\begin{aligned} \begin{bmatrix} \dot{x}_{i[3]} \\ \dot{z}_{i[3]} \end{bmatrix} &= \begin{bmatrix} \beta\mu_i & \omega \\ -\omega & \beta\mu_i \end{bmatrix} \begin{bmatrix} x_{i[3]} - O_{i[3]} \\ z_{i[3]} \end{bmatrix} - \beta r_{i[3]}^2 \begin{bmatrix} x_{i[3]} - O_{i[3]} \\ z_{i[3]} \end{bmatrix} \\ &+ \frac{1}{2} \mathbf{R}(\psi_{i[3]}^{j[1]}) \begin{bmatrix} x_{j[1]} - O_{j[1]} \\ z_{j[1]} \end{bmatrix} \end{aligned}$$

where $r_i[k]$ is the norm of vector $(x_i[k] - O_{i[k]}, z_i[k])^T$ ($k = 1, 3$, that is hip and knee joints) and $i, j = \text{Fore Left, Fore Right, Hind Left and Hind Right limbs (FL, FR, HL, HR)}$. The linear terms are rotated onto each other by the rotation matrices $\mathbf{R}(\theta_{i[1]}^{j[1]})$ and $\mathbf{R}(\psi_{i[3]}^{j[1]})$, where $\theta_{i[1]}^{j[1]}$ is the required relative phase among the $i[1]$'s and $j[1]$'s hip oscillators to perform a walking gait (we exploit the fact that $\mathbf{R}(\theta) = \mathbf{R}^{-1}(-\theta)$) and $\psi_{i[3]}^{j[1]}$ is the required relative phase among the $i[3]$'s and $j[1]$'s CPGs (see Fig. 2). The $\psi_{i[3]}^{j[1]}$ were all set to -90° . Due to the properties of this type of coupling among oscillators, the generated trajectories are always smooth and thus potentially useful for trajectory generation in a robot.

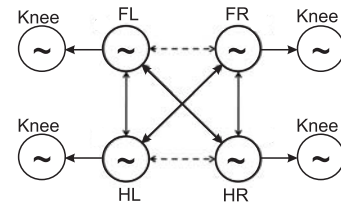


Fig. 2. Locomotion controller architecture depicting coupling structure among the CPGs for a walking gait. The footfall sequence is: HL-FL-HR-FR, with each foot lagging a quarter of a cycle from the previous.

This network structure constitutes the locomotion controller that generates desired trajectories, x_i , obtained by integrating the CPGs dynamical systems. These are sent

online for the PID controllers of each hip and knee joints and result in the actual trajectories \hat{x}_i .

B. Head Movement Generation

Head movement is generated similarly to locomotion, but a CPG for a given DOF is modelled as an Hopf oscillator, not coupled to any other oscillator. Each CPG, therefore, generates a rhythmic movement according to

$$\begin{bmatrix} \dot{x}_i \\ \dot{z}_i \end{bmatrix} = \begin{bmatrix} \beta\mu_i & \omega \\ -\omega & \beta\mu_i \end{bmatrix} \begin{bmatrix} x_i - O_i \\ z_i \end{bmatrix} - \beta r_i^2 \begin{bmatrix} x_i - O_i \\ z_i \end{bmatrix}, \quad (3)$$

where i = tilt, pan, nod.

The control policy is the x_i variable, obtained by integrating the CPGs dynamical systems, and represents tilt, pan and nod joint angles in our experiments. These are sent online for the corresponding PID controllers.

Note that the final movement for each of these joints is a rhythmic motion which amplitude of movement is specified by μ_i , offset by O_i and its frequency by ω .

III. OPTIMIZATION SYSTEM

In order to implement the head motion required to reduce the camera (head) movement locomotion-induced, it is necessary one or several optimal combinations of amplitude, offset and frequency of each head oscillator. This is possible because we can easily modulate amplitude, offset and frequency of the generated trajectories according to changes in the CPG parameters and these are represented in an explicit way by our CPG. Therefore, we have to tune the CPG parameters: amplitude μ_i , offset O_i and frequency ω .

The multitude of parameter combinations is large, and it is difficult to derive an accurate model for the tested quadruped robot and for the environment. Besides, such a model based approach would also require some post-adaptation of results (because of backlash, friction, etc).

In this study, the search of parameters suitable for the implementation of the required head motion was carried out based on the data from a simulated quadruped robot. We recorded the (X, Y, Z) head coordinates in a world coordinate system (Fig. 3), when the robot walks during 30 s and no head stabilization is performed. We are interested in the opposite of this movement around the (X, Y, Z) coordinates. From now on, this data is referred to as $(X, Y, Z)_{\text{observed}}$.



Fig. 3. World coordinate system.

The optimization system runs during 30 s. This is the time for which we have implemented head stabilization. It is arbitrary and could have been chosen differently. We use a GA as a method of optimization of the combinations of the different CPG parameters.

The basic idea is to combine the CPG model for head movement generation with the optimization algorithm. Fig. 4 illustrates a schematics of the overall optimization system. Three head CPGs (3) generate rhythmic motion for the tilt,

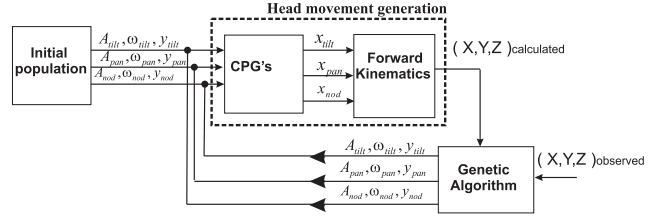


Fig. 4. Schematics of the Optimization system.

pan and nod joints. By applying forward kinematics, we calculate the resulting $(X, Y, Z)_{\text{calculated}}$ head coordinates in the world coordinate system. The distance between the observed and calculated head coordinates is the evaluated criterion used to explore the parameter space of the CPG model to identify the head movement that minimizes the one induced by locomotion itself.

A. Genetic Algorithm

Genetic Algorithms start from a pool of points, usually referred to as chromosomes. Thus, chromosomes represent potential optimal solutions of the problem being solved. In order to implement a GA, it is necessary to define the representation of the search space and a fitness function which permits the comparison between the different chromosomes. Furthermore, genetic operators and the selection mechanism must also be defined.

One or several optimal combinations of amplitude, offset and frequency of each tilt, pan and nod oscillators are necessary in order to generate the desired head movement. Therefore, each chromosome consists in 9 CPG free parameters, as shown in fig 5, that span our vector space for the optimization.

$$\text{Point} \rightarrow \begin{bmatrix} A_{\text{tilt}} & y_{\text{tilt}} & w_{\text{tilt}} & A_{\text{pan}} & y_{\text{pan}} & w_{\text{pan}} & A_{\text{nod}} & y_{\text{nod}} & w_{\text{nod}} \end{bmatrix}$$

Fig. 5. A chromosome is made of nine CPG free parameters.

In our optimization system, we begin the GA search by randomly generating an initial population of chromosomes.

The GA selection operator assures that chromosomes are copied to the next generation with a probability associated to their fitness values. Therefore, this operator mimics the survival of the fittest in the natural world. Although selection assures that in the next generation the best chromosomes will be present with a higher probability, it does not search the space, because it just copies the previous chromosomes. The search results from the creation of new chromosomes from old ones by the application of genetic operators. The crossover operator, takes two randomly selected chromosomes; one point along their common length is randomly

selected, and the characters of the two parent strings are swapped, thus generating two new chromosomes. The mutation operator, randomly selects a position in the chromosome and, with a given probability, changes the corresponding value. This operator does assure that new parts of the search space are explored, which selection and crossover could not fully guarantee.

In this work, real representation of the variables was considered. So, each vector consists of a vector of real values representing the decision variables of the problem. Genetic operators were chosen taking into account this representation. In order to recombine and mutate chromosomes, the Simulated Binary Crossover (SBX) and Polynomial Mutation were considered, respectively. These operators simulate the working of the traditional binary operators [3]. In order to select chromosomes for the application of genetic operators, a tournament selection was implemented.

B. Problem Definition

The distance between the observed and calculated head coordinates is used as fitness function in order to evaluate the resulting head movement. A chromosome giving a smaller distance is evaluated as excellent and is the best chromosome. The fitness of the i chromosome is given by

$$f_i = \sum_{j=1}^n \sqrt{(X_j - X'_j)^2 + (Y_j - Y'_j)^2 + (Z_j - Z'_j)^2} \quad (4)$$

where j is an head position sample (because the points are generated and acquired in a discrete manner), n is total number of samples originated during the evaluation time and (X, Y, Z) , (X', Y', Z') represent observed and calculated head coordinates with the CPG parameters of the i chromosome, respectively.

Only head position errors are computed in the fitness function, because we only control three DOFs and as such cannot control head orientation.

The problem has several simple boundary constraints. In order to guarantee the feasibility of the initial solutions and all solutions generated during the search, a repair mechanism was implemented. Thus, an infeasible solution is repaired exploring the relations among variables expressed by the constraints. In this case, the search ranges of the CPG parameters were set beforehand as shown in Table I for the purpose of efficient learning and according to the limits of the tilt, pan and nod DOFs.

Search for optimal parameters is carried out by performing the overall optimization process over a preset number of generations.

C. Results

We depict results when a population was established with 100 chromosomes and a preset number of 300 generations was set. The evaluation time, for head movement generation is 30 s.

Fig. 6 shows the fitness evolution of the best point for each of the 300 generations. The best point has a fitness value of 3972 that was achieved at iteration 179.

TABLE I
SEARCH RANGES OF CPG PARAMETERS

	Lower limit	Upper limit
$A_{\text{tilt}}(^{\circ})$	-75	0
$y_{\text{tilt}}(^{\circ})$	$-75 + \frac{A_{\text{tilt}}}{2}$	$0 - \frac{A_{\text{tilt}}}{2}$
$w_{\text{tilt}}(\text{rad s}^{-1})$	1	12
$A_{\text{pan}}(^{\circ})$	-88	88
$y_{\text{pan}}(^{\circ})$	$-88 + \frac{A_{\text{tilt}}}{2}$	$88 - \frac{A_{\text{tilt}}}{2}$
$w_{\text{pan}}(\text{rad s}^{-1})$	2	12
$A_{\text{nod}}(^{\circ})$	-15	45
$y_{\text{nod}}(^{\circ})$	$-15 + \frac{A_{\text{tilt}}}{2}$	$45 - \frac{A_{\text{tilt}}}{2}$
$w_{\text{nod}}(\text{rad s}^{-1})$	1	12

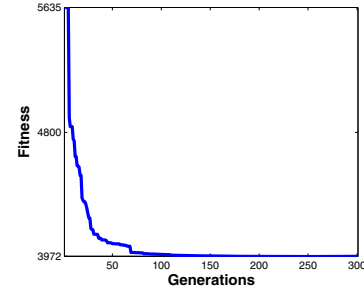


Fig. 6. Fitness evolution of the best point for each of the 300 generations.

Table II shows tuned CPG parameters representing the best point of the 1st and 300th generation.

TABLE II
BEST POINT CPG PARAMETERS

	1st generation	300th generation
$A_{\text{tilt}}(^{\circ})$	0.3	1.8234
$y_{\text{tilt}}(^{\circ})$	0	-0.9117
$w_{\text{tilt}}(\text{rad s}^{-1})$	4.188	4.1120
$A_{\text{pan}}(^{\circ})$	5.5	8.0461
$y_{\text{pan}}(^{\circ})$	0	0.1232
$w_{\text{pan}}(\text{rad s}^{-1})$	2.094	2.1304
$A_{\text{nod}}(^{\circ})$	0.3	0.4955
$y_{\text{nod}}(^{\circ})$	0	-1.1609
$w_{\text{nod}}(\text{rad s}^{-1})$	4.188	3.9968

Fig. 7 allows a better understanding of the evaluation of the fitness function, and represents for each best solution in each generation, the distance between observed and calculated values of head movement in each sample time of the evaluation time, . In the figure we can observe that this distance is smaller for generation 300. In fig 7, the variation of d_j is shown for the best solutions of 1st (dotted line) and 300th (solid line) generations for time ranging between $t = 5$ and 15 s. In the figure we can observe that this distance is smaller for generation 300.

Fig. 8 depicts the time courses of the (X, Y, Z) calculated (solid line) and observed (dotted line) head movement according to the CPG parameters of the final best solution. This data was mathematical treated such as to keep only the

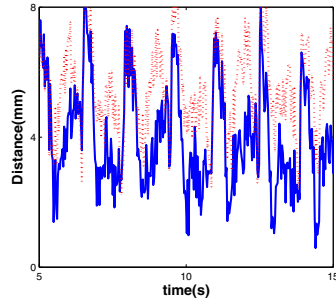


Fig. 7. Distance for generations 1 (dotted line) and 300 (solid line) and time between 5 and 15 seconds.

oscillations in the movement and remove the drift that the robot has in the X coordinate and also the forward movement in the Z coordinate. Table III gives the maximal movement variation in the (X, Y, Z) coordinates for the observed and calculated movement. We conclude that the generated movements are quite similar in the X coordinate. The calculated movement is quite different in the Z coordinate. This results from the fact that the pan joint controls movement in the X coordinate, while both the tilt and nod joints control the Y and Z coordinates. Thus, if we are able to achieve similar movements in the Y coordinate, this is more difficult to achieve in the Z coordinate.

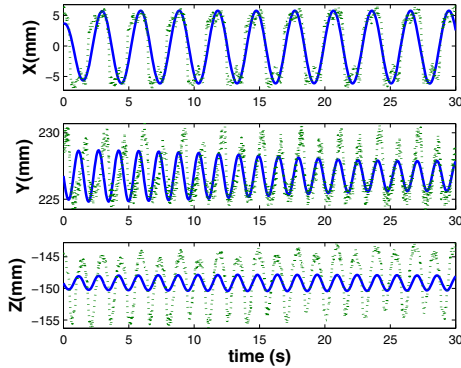


Fig. 8. (X, Y, Z) calculated (solid line) and observed (dotted line) head movement according to the CPG parameters of the final best point.

TABLE III
MAXIMAL MOVEMENT VARIATION IN (X, Y, Z)

	max variation X(mm)	max variation Y(mm)	max variation Z(mm)
Observed Movement	13.42	5.9	11.3
Calculated Movement	12.02	3.7	8.5

Fig 9 depicts 3D observed and calculated head movement for the final best point. In this figure we confirm that these two movements are quite similar and main differences are observed at the Z coordinate.

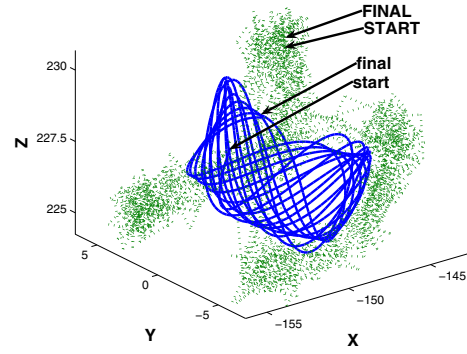


Fig. 9. 3D observed (dotted line) and calculated (solid line) head movement according to the CPG parameters of the final best point. START and start indicate where the observed and calculated movement started, respectively. The FINAL and the final indicate where the observed and calculated movement ended, respectively.

IV. SIMULATION RESULTS

Our aim was to build a system able to reduce the head motion of a robot that walks in the environment. For that, we set a dynamical controller generating trajectories for the head joints such that the final head movement is opposite to the one induced by locomotion.

In this section, we describe the experiment done in a simulated ers-7 AIBO robot using Webots [9]. These experiments can be completely seen in the attached video. Webots is a software for the physic simulation of robots based on ODE, an open source physics engine for simulating 3D rigid body dynamics. We simulate the exact number of DOFs, mass distributions and the visual system.

The locomotion controller generates the joint angles of the hip and knee joints in the sagittal plane, that is 8 DOFs of the robot, 2 DOFs in each leg. Only walk gait is generated and tested.

The head controller generates the joint angles of the 3 DOFs: tilt, pan and nod. The other DOFs are not used for the moment, and remain fixed to an appropriately chosen value during the experiments. The AIBO has a camera built into its head.

At each sensorial cycle (30 ms), sensory information is acquired. The dynamics of the CPGs are numerically integrated using the Euler method with a fixed time step of 1 ms thus specifying servo positions. Parameters were chosen in order to respect feasibility of the experiment and are given in table IV and V.

TABLE IV
PARAMETER VALUES FOR GENERATING LOCOMOTION

	β	ω (rad s^{-1})	μ_1	$\frac{1}{2\beta\mu_1}$ (s)
Front Limbs	0.1	2.044	6.25	0.8
Hind Limbs	0.025	2.044	25	0.8

TABLE V
PARAMETER VALUES FOR GENERATING HEAD MOTION

	β	ω (rad s ⁻¹)	μ_i	$\frac{1}{2\beta\mu_i}$ (s)
<i>tilt</i>	0.75	4.19	0.83	0.8
<i>pan</i>	0.04	2.09	16.18	0.8
<i>nod</i>	10.18	4.19	0.06	0.8

Because we are working in a simulated environment, we are able to build a GPS into the AIBO camera, that enable us to verify how the head effectively moves in an external coordinate system. Two simulations are performed: the robot walks during 30s with and without the feedforward solution and its GPS coordinates are recorded. Results are compared for these two simulations. Fig. 10 shows the GPS coordinates for the experiments with (solid line) and without the feedforward solution (dotted line).

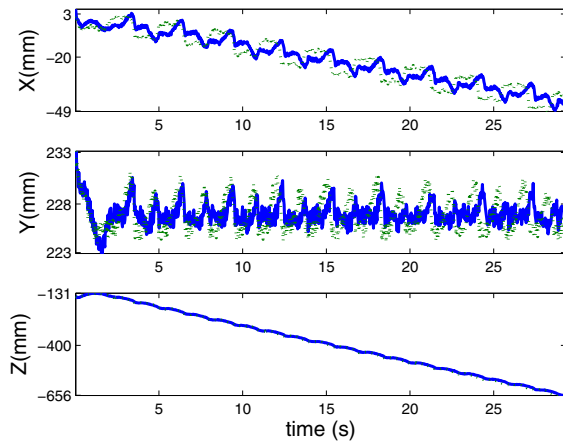


Fig. 10. (X, Y, Z) coordinates of the GPS positioned in the AIBO head when the robot walks during 30s. Solid and dotted lines indicate the experiment in which the feedforward solution is and is not implemented, respectively.

We expect that the proposed feedforward solution minimizes the variation of the GPS coordinates, meaning that the head remains near the same position during the experiment. We clearly observe that the X and Y coordinates have smaller variations. Note that there is some drift in the X coordinates, meaning the robot slightly deviates towards its side while walking. The observed peaks in the Y coordinate reflect the final stage of the swing phase and the begin of the stance phases of the fore legs, corresponding to an accentuated movement of the robot center of mass. This problem will be addressed in current work, by improving the locomotion controller and take into account balance control [4].

As expected, we verify that the proposed feedforward solution minimizes the head movement. This minimization is mostly verified in the X and Y coordinates as previously explained.

V. CONCLUSIONS AND FUTURE WORKS

In this article, we have addressed head stabilization of a quadruped robot that walks with a walking gait.

A locomotion controller based on dynamical systems, CPGs, generates quadruped locomotion. The required head motion needed to reduce the head shaking induced by locomotion, is generated by CPGs built-in in the tilt, pan and nod joints. These CPG parameters are tuned by an optimization system. This optimization system combines CPGs and a genetic algorithm. As a result, sets of parameters obtained by the evolution strategy were adequate for the implementation of a head movement that does reduce the one induced by the locomotion.

Currently, we are using other optimization methods such as particle swarm [11] and electromagnetism algorithm. We will extend this optimization work to address other locomotion related problems, such as: the generation and switch among different gaits according to the sensorial information and the control of locomotion direction. We further plan to extend our current work to online learning of the head movement similarly to [10].

REFERENCES

- [1] S Degallier, C P. Santos, L Righetti and A Ijspeert, Movement Generation using Dynamical Systems: a Drumming Humanoid Robot. *In Humanoids'06 IEEE-RAS International Conference on Humanoids Robots*, Genova, Italy, December 4-6 2006
- [2] Goldberg, D., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, (1989).
- [3] Deb, K. Agrawal, R.B., Simulated binary crossover for continuous search space. *Complex Systems* 9(2), 115-149, (1995).
- [4] Castro, Luiz, Santos, C P; Oliveira, M and Ijspeert, A, Postural Control on a Quadruped Robot Using Lateral Titl: a Dynamical System Approach, *European Robotics Symposium EUROS 2008*, Prague, 2008.
- [5] Takizawa, S, Ushida, S, Okatani, T, Deguchi, K., 2DOF Motion Stabilization of Biped Robot by Gaze Control Strategy. *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, Aug. 2005:3809-3814
- [6] Ravi Kaushik, Marek Marcinkiewicz, Jizhong Xiao, Simon Parsons, and Theodore Raphan, Implementation of Bio-Inspired Vestibulo-Ocular Reflex in a Quadrupedal Robot, *2007 IEEE International Conference on Robotics and Automation (ICRA 2007)*, Roma, Italy, 10-14 April 2007: 4861-4866
- [7] Yamada, H.; Mori, M.; Hirose, S., Stabilization of the head of an undulating snake-like robot, *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007)*, San Diego, CA, USA, Oct 29 - Nov 2, 2007 : 3566-3571
- [8] Yurak Son, Takuya Kamano, Takashi Yasuno, Takayuki Suzuki, and Hironobu Harada, Generation of Adaptive Gait Patterns for Quadruped Robot with CPG Network Including Motor Dynamic Model, *Electrical Engineering in Japan*, Vol. 155, No. 1, 2006
- [9] O. Michel, "Webots: Professional mobile robot simulation," *Int. Journal of Advanced Robotic Systems*, vol. 1, no. 1, pp. 39-42, 2004.
- [10] A. Sprowitz, R. Moeckel, J. Maye, A. Ijspeert, Learning to Move in Modular Robots using Central Pattern Generators and Online Optimization; *The Int. Journal of Robotics Research* 27, 423-443, 2008.
- [11] J. Kennedy and R.C. Eberhart, *Particle swarm optimization*, in *IEEE International Conference on Neural Network*, 1995, pp. 1942-1948.
- [12] S. Grillner, Neurobiological bases of rhythmic motor acts in vertebrates. *Science*, Vol. 228, No. 4696, pp. 143-149, 1985
- [13] J Pratt, C Chew, A Torres, P Dilworth, G Pratt, Virtual Model Control, An intuitive approach for bipedal locomotion, *The Int. J. of Robotics Research*, 2001, 20 (2), 129-143